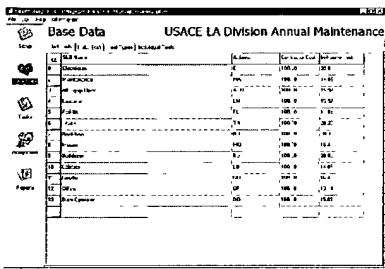
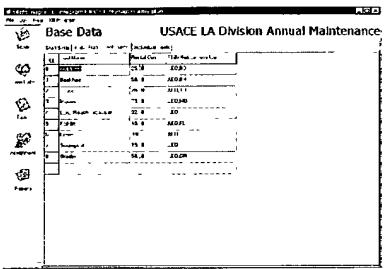
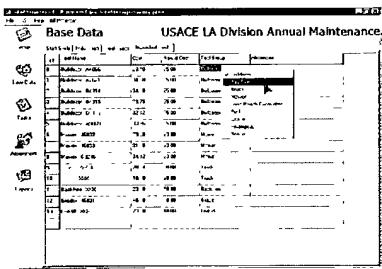


 <b>Setup</b>  <b>Default Values</b>  <b>Time Information</b>  <b>Project Information</b>  <b>Default Path</b>  <b>Help</b>													
<table border="1"> <tr> <td>Time Zone:</td> <td>EST (Eastern Standard Time)</td> </tr> <tr> <td>Daylight Savings:</td> <td>ON (Daylight Savings Time)</td> </tr> <tr> <td>Server Name:</td> <td>Server for USACE LA Division</td> </tr> <tr> <td>Event Start Date:</td> <td>1999-01-01</td> </tr> <tr> <td>Event End Date:</td> <td>1999-01-01</td> </tr> </table>		Time Zone:	EST (Eastern Standard Time)	Daylight Savings:	ON (Daylight Savings Time)	Server Name:	Server for USACE LA Division	Event Start Date:	1999-01-01	Event End Date:	1999-01-01		
Time Zone:	EST (Eastern Standard Time)												
Daylight Savings:	ON (Daylight Savings Time)												
Server Name:	Server for USACE LA Division												
Event Start Date:	1999-01-01												
Event End Date:	1999-01-01												
<table border="1"> <tr> <td>Week Starts On:</td> <td>Sunday</td> </tr> <tr> <td>Week End On:</td> <td>Saturday</td> </tr> <tr> <td>Month Starts On:</td> <td>January</td> </tr> <tr> <td>Month End On:</td> <td>December</td> </tr> <tr> <td>Year Starts On:</td> <td>January</td> </tr> <tr> <td>Year End On:</td> <td>December</td> </tr> </table>		Week Starts On:	Sunday	Week End On:	Saturday	Month Starts On:	January	Month End On:	December	Year Starts On:	January	Year End On:	December
Week Starts On:	Sunday												
Week End On:	Saturday												
Month Starts On:	January												
Month End On:	December												
Year Starts On:	January												
Year End On:	December												

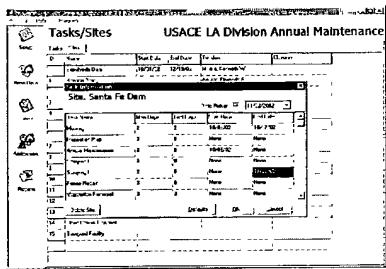


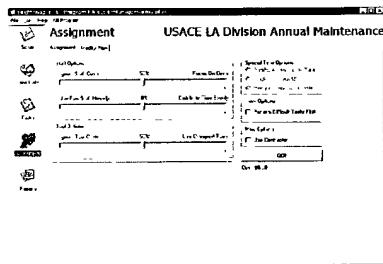


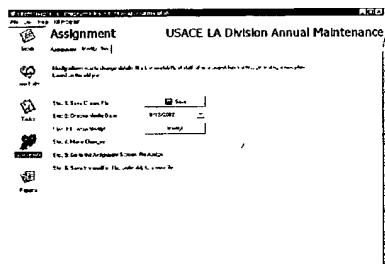




Task/Site	Task ID	Task Description	Task Status	Task Due Date	Task Last Update
1. Annual Inspection	1	Annual Inspection	Planned	2023-06-30	2023-05-15
2. Annual Training	2	Annual Training	Planned	2023-07-15	2023-05-15
3. Annual Audit	3	Annual Audit	Planned	2023-08-01	2023-05-15
4. Annual Audit	4	Annual Audit	Planned	2023-08-01	2023-05-15
5. Annual Audit	5	Annual Audit	Planned	2023-08-01	2023-05-15
6. Annual Audit	6	Annual Audit	Planned	2023-08-01	2023-05-15
7. Annual Audit	7	Annual Audit	Planned	2023-08-01	2023-05-15
8. Annual Audit	8	Annual Audit	Planned	2023-08-01	2023-05-15
9. Annual Audit	9	Annual Audit	Planned	2023-08-01	2023-05-15
10. Annual Audit	10	Annual Audit	Planned	2023-08-01	2023-05-15



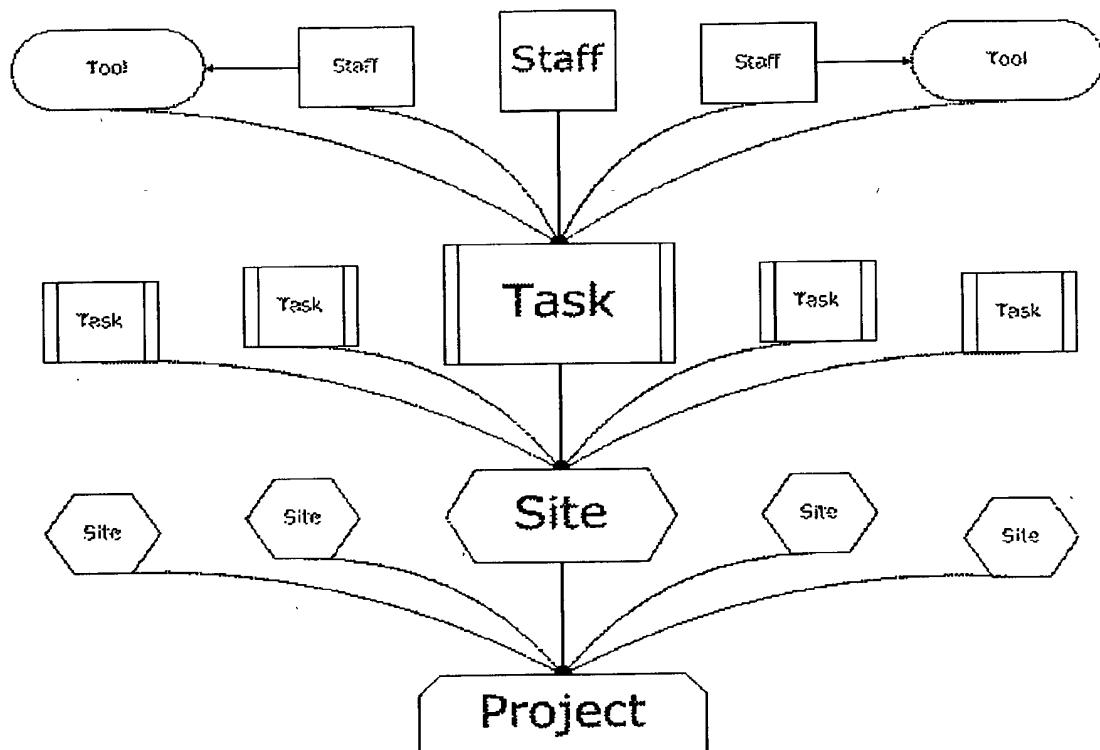




Task Name	Duration	Start	29.02	Oct 02							
Repaireds Dam	10-0-02	Tue 10/1/02									
Mowing	2 days	Tue 10/1/02									
Mowing II	2 days	Tue 10/3/02									
Mowing III	2 days	Mon 10/7/02									
Inspectio Prep	2 days	Tue 10/7/02									
Assit Maintance	2 days	Tue 10/3/02									
Spraying I	2 days	Mon 10/7/02									
Spraying II	1 day	Wed 10/9/02									
Fence Repair	2 days	Tue 10/10/02									
Vegetable Removal	3 days	Tue 10/14/02									
Electrical Repair	1 day	Mon 10/14/02									
Mechanical Maintenance	2 days	Tue 10/15/02									
Outrane	1 day	Tue 10/17/02									
Equipment Operators	3 days	Tue 10/17/02									
Ramse Fd Dam	15-4 days	Tue 10/17/02									
Mowing	2 days	Wed 10/9/02									
Mowing II	4 days	Fri 10/11/02									
Mowing III	2 days	Tue 10/15/02									
Inspectio Prep	2 days	Tue 10/15/02									
Assit Maintance	2 days	Tue 10/3/02									
Spraying I	2 days	Mon 10/7/02									
Spraying II	1 day	Tue 10/17/02									
Fence Repair	4 days	Fri 10/18/02									
Vegetable Removal	5 days	Fri 10/4/02									
Electrical Repair	1 day	Tue 10/22/02									
Mechanical Maintenance	2 days	Wed 10/23/02									
Outrane	1 day	Fri 10/25/02									
Equipment Operators	3 days	Wed 10/25/02									
Ramse Fd Dam	20-3 days	Tue 10/1/02									
Mowing	4 days	Fri 10/16/02									
Mowing II	2 days	Tue 10/22/02									
Mowing III	2 days	Tue 10/24/02									

 US Army Corps of Engineers Los Angeles District	<b>Task</b> Split Progress	<b>Milestone</b> Summary Project Summary	<b>External Tasks</b> External Milestone Deadline

## Relationship Between Staff/Tool/Task/Site



```

// This class allows for sorting based on scores
// without losing track of the original object
class ObjScore
{
public:
    int id;
    int score;
}

void GetStaffList(Task* t, Array<Personnel*> &list)
{
    // Get All the Staff who can do this task
    // and put it in the array "list"
    GetAllSkilledStaff(t, list)

    // Make an array of score objects
    // the size of "list"
    Array<ObjScore> scores(list.length())

    for(i=0;i<list.length();i++)
    {
        // Give this person's ID to the score object
        scores[i].id = list[i]->id

        // Make this score affected by the staff's cost
        // It is negative to show that the higher the cost,
        // the worse the score. Multiply it by some factor
        // based on the specified criteria
        scores[i].score = -(factor) * list[i]->cost

        // Make this score affected by the number of days
        // worked by the staff already. This is multiplied
        // by a work-distribution factor. If the user chooses
        // to spread work evenly, then the less-worked staff
        // will be given a higher score
        scores[i].score += (factor) * DaysWorked(list[i])
    }

    // Sort the list based on score. This uses the quicksort
    // sorting method, with modifications to take a list of
    // ObjScore and not numbers

    OSQuickSort(os,0,os.length()-1)

    // Now make the list reflect the newly ordered list of
    // scores
    for(i=0;i<list.length();i++)
        list[i] = GetStaffFromID(scores[i].id)
}

```

```

// Shows how these resources are inter-connected

// Basic Resource Category
class Resource
{
public:
    int id;                                // ID number to identify resource
    TString name;                          // Name of Tool / Staff
    TArrayList<Task*> tasks;              // List of Tasks and days
    float cost;                            // Resource Rate
    bool useresc;                          // Is this resource available
    TArrayList<Absence*> abs;              // List of absence Dates

    Resource();
};

// Tool Resource
class Tool : public Resource
{
public:
    TArrayList<int*> tool_ids;            // List of tool categories this
                                            // tool falls into
};

// Staff Resource
class Personnel : public Resource
{
public:
    TArrayList<string*> skills;           // Skills this staff has
    TString contact, comment;             // Contact Info and comments
    TArrayList<Resource*> rendering;      // Which sites/dates this staff
                                            // renders for
    float overtime;                      // Overtime Rate

    Personnel();
};

class Site
{
public:
    TString name;                          // Name of Site
    int id;                                // Site ID (For searching/saving)
    TArrayList<Group*> groups;             // Groups (of Tools and Staff) that
                                            // can work at this site
    TArrayList<Personnel*> tenders;          // Possible Site tenders
    TArrayList<Site*> sub;                 // Sub-sites of this site
    TArrayList<TaskPerson*> curtender;       // Tenders and Dates of rendering

    bool closed;                           // Is this site closed down forever
    TArrayList<Absence*> abs;              // Site closures

    TArrayList<SiteTask*> st;              // Site/Task Linkage
};

class Task
{
public:
    SiteTask* st;                          // Site/Task Linkage
    Site* site;                            // Site where this Task takes place
    TArrayList<TaskPerson*> personnel;       // List of staff/days working on
    TArrayList<TaskOvertime*> ots;           // List of overtime start
                                            // List of tool s/days
    int start, end;                        // Task's start and end dates
};

Task();
};

```